



**SWEETWATER COUNTY
SCHOOL DISTRICT #1**

COMPUTER SCIENCE EDUCATION

K-12 CURRICULUM MAP

JUNE 2022

Table of Contents

Acknowledgements	3
District and Subject Mission Statements	4
SCSD#1 Curriculum Terminology	5
How to Read the Computer Science Curriculum Map	6
Computer Science Curriculum at a Glance	7
Computer Science Curriculum Maps:	
Kindergarten Computer Science	9
1st Grade Computer Science	11
2nd Grade Computer Science	14
3rd Grade Computer Science	16
4th Grade Computer Science	18
5th Grade Computer Science	21
6th Grade Computer Science	24
Introduction to Computer Science (7-8)	26
Computer Science 1	28
Webpage Design 1	31
Mobile App Development	33
Cybersecurity	36
Computer Science 2	38
Appendices	
A – Pacing Guide	42

Acknowledgements

Special thanks go to the members of the **Computer Science Subject Area Committee** for their hard work in creating this curriculum:

Michael Alton
Candy Bedard
John Beitler
Jill Blazovich
Hope Downs Lewis
Brenna Franklin
Laura Grossnickle
Emily Jackson

Theresa Lembke
Kalista Pendleton
Michelle Phillpot
Darcie Punches
Crystal Richards
Corrina Searle
Sarah Wright

Also acknowledged are the members of the **Community Curriculum Council**:

Jennifer Allen
Sally Allen
Lorna Bath
Linda Carter
Maggie Coletti
Jason Doer
Hope Downs-Lewis
Carrie Ellison
Tonya Follum

Rhonda Gamble
Jodie Garner
Stacey Hanson
Max Mikkelsen
Michael Moore
Cheryl Notman
Cathy Perkins
Scott Pies
Lisa Plant

Darcie Punches
Barbara Rezzonico
Whitney Sorenson
Ronald Urbin
Debbie Verras
Nathan Wonnacott
Erin Worthington

Sweetwater County School District #1 Vision Statement

As an innovative district, united with our community, we empower and inspire ALL students to academic excellence in pursuit of their interests and passions.

Sweetwater County School District #1 Mission Statement

To provide a quality education for ALL students. The district will accomplish this by:

- making students our first priority
- utilizing community partnerships
- promoting professional excellence
- being committed to excellence in education
- providing a safe, orderly and efficient environment for learning

Computer Science Mission Statement

Students in Sweetwater County School District #1 will utilize principles of computer science to apply computational thinking, critical thinking, and problem-solving skills, both individually and collaboratively. Through real-world applications, they will develop skills to prepare for their technological futures in both local and global societies.

Sweetwater County School District No. 1 Curriculum Terms

Curriculum Term	Definition
Community Curriculum Council (CCC)	advisory council responsible for evaluating current systems and making recommendations regarding curriculum, instruction, and assessment practices
Subject Area Committee (SAC)	team of representatives from a specific subject area who will write the curriculum and common assessments
Curriculum map	what SCSD1 values and guarantees that students will learn
Purpose statement	identifies the purpose of a class
Benchmark	overall outcome for a unit
Learning target	individual skills that lead up to achieving the benchmark
Resource, textbook, program, etc.	resource adopted by the district to help teach the local curriculum
Pacing Guide	identifies when a benchmark will be taught and when it will be assessed
Proficiency Scale	a tool to show learning goals and the progression of learning for students.
Instructional Planning Resources (IPR)	organizational tool for planning lessons based on learning targets rather than days
Formative assessment	informal assessment used to direct instruction
Common Assessment	common assessment given within a benchmark by all teachers who teach the same class

How to Read the Computer Science Curriculum Map

Purpose Statement identifies the purpose of a class and what is new or different at this level.

Purpose Statement:	Students will evaluate the safety and sanitation of a kitchen, show proper technique of kitchen tools and equipment and evaluate food choices using MyPlate guidelines.
--------------------	---

Quarter 1		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
IC.SI.01- Practice grade level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior.		Commonsense.org Lesson 2 Lesson 4 Lesson 6 Lesson 7
NI.C.01- Explain what authentication factors (e.g. login) are, why we use them, and apply authentication to protect devices and information (personal and private) from unauthorized access.	Observational (Daily device use)	Commonsense.org Lesson 3
IC.C.01- Describe how people use different types of technologies in their daily work and personal lives.	SS.2.4.3	Commonsense.org Lesson 5

WY Computer Science Standard
Standards in bold represent the WY Performance Standards and are considered essential.

Suggested Integration
Cross-curricular connection in which this standard may easily fit.

Suggested Lesson
May be used to both learn and teach this standard.

Computer Science Curriculum at a Glance

Grade Level or Course	Purpose Statement
Kindergarten	Students will demonstrate proper use and care of devices and applications. Proper use is shown through turning the device on and off, logging on and off, and selecting various apps. They will also explain how to keep their property and information safe.
1st Grade	Students will follow step-by-step directions to complete tasks. They will also model and explain what authentications factors are and why they are important for a digital citizen. Students will identify and problem solve basic errors associated with devices and apps.
2nd Grade	Students will write basic step-by-step directions to create algorithms and programs to accomplish tasks. Students will interact respectfully with others while making appropriate choices when participating in an online community.
3rd Grade	Students will identify and describe the purpose of the physical workings of a computer and how they communicate in a network. Students will explain and demonstrate how we stay safe in a digital environment.
4th Grade	Students will explain how we use computers to store and organize data and justify where they store information and for what purpose. Students will explain how we use computers ethically, how to protect information, and demonstrate appropriate behavior when interacting with classmates in a digital environment.
5th Grade	Students will apply computational thinking skills to create and build basic programs to accomplish a task and debug issues to ensure that the program runs as intended.
6th Grade	Students will apply computational thinking to identify and resolve problems with devices. Students will investigate and select computational tools to create data for analysis. Students will critique the use of computing in society and develop strategies for safe and ethical use.
Introduction to Computer Science (7-8)	Students will work independently and collaboratively to collect and transform grade appropriate content, find and manipulate data, make the data more useful and secure. Students will also create and improve (test and debug) a digital program.
Computer Science	Students will demonstrate the basics of hardware and software and explain their interactions. Students will build and strengthen logical thinking and problem-solving skills as students develop the

	foundations of computer programming (in either block or text coding) and create programs to solve everyday problems.
Webpage Design 1	Students will analyze the structure of existing websites and use HTML and CSS to create custom websites that will display information and include interactive user interfaces. Students will test multiple prototypes that address accessibility for a variety of users.
Mobile App Development	Students will design and program mobile apps for multiple platforms that address current problems/needs. Students will include interactive user interfaces with accessibility features for diverse users into their designs.
Cybersecurity	Students will demonstrate principles of digital citizenship and cyber hygiene. Students will develop skills in cryptography, software security, networks and IT infrastructure. Students will recommend various security measures, compare and explain trade-offs and ethical impacts.
Computer Science 2	Students will categorize the roles of operating system software and explain how computers facilitate logic, input, output and storage. Students will build and strengthen logical thinking and problem-solving skills as they develop skills of computer programming and create programs to solve everyday problems. Students will analyze metadata from programs and create visual representations to find patterns and explain how data is used to make decisions and learn about our world.

Kindergarten Computer Science

Purpose Statement:	Students will demonstrate proper use and care of devices and applications. Proper use is shown through turning the device on and off, logging on and off, and selecting various apps. They will also explain how to keep their property and information safe.
--------------------	---

Bolded items identify learning targets/standards that must be taught to mastery and are considered a priority. Please note, however, that all learning targets/standards in the curriculum map must still be taught, but those in bold should be given extra emphasis.

Quarter 1		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
CS.D.01 - Independently select and use a computing device to perform a variety of tasks for an intended outcome (e.g. create an artifact). <i>Explanation: Device usage - students can turn computer/ipad on/off, log on/off, select an app.</i>	Observational (Daily device use)	N/A
CS.HS.01 - Demonstrate and describe the function of common components of computing systems (hardware and software) (e.g. use a browser, search engine). <i>Explanation: Students will access a search engine and describe what the purpose of that is.</i>	Observational (Daily device use)	N/A
<i>Vocabulary: on, off, power, open, close, home button, select, click, online, app, website</i>		

Quarter 2		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
IC.SI.01 - Practice grade level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior.	SS.K.4.2 SS.K.2	Code.org Lesson 1 Unplugged
NI.C.01- Explain what authentication factors (e.g. login) are, why we use them, and apply authentication to protect devices and information (personal and private) from unauthorized access.	Observational (Daily device use)	N/A
<i>Vocabulary: safe, kind, responsible, respectful, private, public, personal, information, identity, password, username, login</i>		

Quarter 3		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
AP.M.01- Using grade appropriate content and complexity, decompose (breakdown) the steps needed to solve a problem into a precise sequence of instructions (e.g. develop a set of instructions on how to play your favorite game). <i>(Assessed Quarter 4)</i>		Code.org Lesson 2
<i>Vocabulary:</i> on, off, power, open, close, home button, select, click, online, app, website		

Quarter 4		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
AP.A.01 - With guidance, identify and model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete tasks (e.g. verbally, kinesthetically, with robot devices, or a programming language).	ELAK.6.1 SS.K.6 W.K.3	Code.org Lesson 3 unplugged
AP.M.01 - Using grade appropriate content and complexity, decompose (breakdown) the steps needed to solve a problem into a precise sequence of instructions (e.g. develop a set of instructions on how to play your favorite game).	SCI.K.3	Code.org Lesson 4 Lesson 5 Lesson 6
<i>Vocabulary:</i> algorithm, bug, debugging, program, programming, click, double-click, drag, drop		

1st Grade Computer Science

Purpose Statement:	Students will follow step-by-step directions to complete tasks. They will also model and explain what authentications factors are and why they are important for a digital citizen. Students will identify and problem solve basic errors associated with devices and apps.
--------------------	---

Bolded items identify learning targets/standards that must be taught to mastery and are considered a priority. Please note, however, that all learning targets/standards in the curriculum map must still be taught, but those in bold should be given extra emphasis.

Quarter 1		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
CS.D.01- Independently select and use a computing device to perform a variety of tasks for an intended outcome (e.g. create an artifact). <i>Explanation: Device usage - students can turn computer/ipad on/off, log on/off, select an app/program.</i>	Observational (Daily device use)	Code.org Lesson 10 Unplugged
CS.HS.01- Demonstrate and describe the function of common components of computing systems (hardware and software) (e.g. use a browser, search engine). <i>Explanation: Students will access a search engine and describe what the purpose of that is.</i>	Observational (Daily device use)	N/A
CS.T.01- Recognize computing systems might not work as expected and identify and effectively communicate simple hardware or software problems and implement solutions (e.g. app or program is not working as expected, no sound is coming from device, caps lock turned on) and discuss problems with peers and adults. <i>(Assessed Quarter 4)</i>	Observational (Daily device use)	N/A
IC.SI.01- Practice grade level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior.	Observational (Daily device use) SS.1.2	Code.org Lesson 1 Unplugged
<i>Vocabulary: on, off, power, open, close, home button, select, click, online, app, website, digital footprint, permanent, private information, trail</i>		

Quarter 2		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
AP.M.01- Using grade appropriate content and complexity, decompose (breakdown) the steps needed to solve a problem into a precise sequence of instructions (e.g. develop a set of instructions on how to play your favorite game).	1.W.3	Code.org Lesson 2 Unplugged Lesson 3 Lesson 4 Lesson 5
NI.NCO.01- Identify and <i>recognize</i> that computing devices can be connected in a variety of ways (e.g. Bluetooth, Wi-Fi, home and school networks, the internet). <i>Explanation: Students will identify that there are different ways to connect their devices. They will recognize the different connectivity options (Bluetooth, internet).</i>	Observational (Daily device use)	N/A
NI.C.01- Explain what authentication factors (e.g. login) are, why we use them, and apply authentication to protect devices and information (personal and private) from unauthorized access.	Observational (Daily device use)	N/A
CS.T.01- Recognize computing systems might not work as expected and identify and effectively communicate simple hardware or software problems and implement solutions (e.g. app or program is not working as expected, no sound is coming from device, caps lock turned on) and discuss problems with peers and adults. <i>(Assessed Quarter 4)</i>	Observational (Daily device use)	N/A
<i>Vocabulary: private, public, personal, information, identity, password, username, login, select, click, online, app, website, algorithm, bug, debugging, program, programming, persistence, laptop, cell phone, ipad, device, Bluetooth, WIFI, network, internet</i>		

Quarter 3		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
AP.A.01 - With guidance, identify and model daily processes by creating and following algorithms (sets of step-by-step instructions) to complete tasks (e.g. verbally, kinesthetically, with robot devices, or a programming language).	1.W.3 SS.1.6	Code.org Lesson 6 Unplugged Lesson 7 Lesson 8 Lesson 9

AP.V.01- Model the way programs store and manipulate data by using numbers or other symbols to represent information (e.g. thumbs up/thumbs down as representations of yes/no, arrows when writing algorithms to represent direction, or encode and decode words using numbers, pictographs, or other symbols to represent letter or words).	SS.1.6	
CS.T.01- Recognize computing systems might not work as expected and identify and effectively communicate simple hardware or software problems and implement solutions (e.g. app or program is not working as expected, no sound is coming from device, caps lock turned on) and discuss problems with peers and adults. (Assessed Quarter 4)	Observational (Daily device use)	-----
Vocabulary: loop, repeat, symbols, representation		

Quarter 4		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
AP.M.01- Using grade appropriate content and complexity, decompose (breakdown) the steps needed to solve a problem into a precise sequence of instructions (e.g. develop a set of instructions on how to play your favorite game).		Code.org Lesson 11 Unplugged Lesson 12
CS.T.01- Recognize computing systems might not work as expected and identify and effectively communicate simple hardware or software problems and implement solutions (e.g. app or program is not working as expected, no sound is coming from device, caps lock turned on) and discuss problems with peers and adults. (Assessed Quarter 4)	Observational (Daily device use)	N/A
Vocabulary: event		

2nd Grade Computer Science

Purpose Statement:	Students will write basic step-by-step directions to create algorithms and programs to accomplish tasks. Students will interact respectfully with others while making appropriate choices when participating in an online community.
--------------------	--

Bolded items identify learning targets/standards that must be taught to mastery and are considered a priority. Please note, however, that all learning targets/standards in the curriculum map must still be taught, but those in bold should be given extra emphasis.

Quarter 1		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
IC.SI.01- Practice grade level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior.		Commonsense.org- Lesson 2 Lesson 4 Lesson 6 Lesson 7
NI.C.01- Explain what authentication factors (e.g. login) are, why we use them, and apply authentication to protect devices and information (personal and private) from unauthorized access.	Observational (Daily device use)	Commonsense.org- Lesson 3
IC.C.01- Describe how people use different types of technologies in their daily work and personal lives.	SS.2.4.3	Commonsense.org- Lesson 5

Quarter 2		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
DA.S.01 - With guidance develop and modify an organizational structure by creating, copying, moving, and deleting files and folders.	Google drive	N/A
DA.CVT.01 - With guidance collect data and independently present the same data in various visual formats.	M.2.8	Code.org Lesson Lesson 14
DA.IM.01- With guidance, interpret data and present it in a chart or graph (visualization) in order to make a prediction with or without a computing device.	M.2.8	N/A

Quarter 3		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
AP.C.01 - With guidance independently and collaboratively create programs to accomplish tasks using a programming language, robot device, or unplugged activity that includes sequencing, conditionals, and repetition.		Code.org Lesson 2
AP.PD.01 - Develop plans that describe a programs sequence of events, goals, and expected outcomes.		Code.org Lesson 3
AP.PD.02 - Give credit to ideas, creations, and solutions of others while writing and developing programs.		commonsense.org Lesson 7

Quarter 4		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
AP.PD.03 - Independently and collaboratively debug (identify and fix errors) programs using a programming language.		Code.org Lesson 4
AP.PD.04- Use correct terminology (debug, program input/output, code) to explain the development of a program or an algorithm (e.g. in an unplugged activity, hands-on manipulatives, or a programming language).		Code.org Lesson 5

3rd Grade Computer Science

Purpose Statement:	Students will identify and describe the purpose of the physical workings of a computer and how they communicate in a network. Students will explain and demonstrate how we stay safe in a digital environment.
--------------------	--

Bolded items identify learning targets/standards that must be taught to mastery and are considered a priority. Please note, however, that all learning targets/standards in the curriculum map must still be taught, but those in bold should be given extra emphasis.

Quarter 1		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.CS.D.01 - Independently, describe how internal and external parts of computing devices function to form a system.	Science - parts of a system Input/Process/Output	Code.org video
5.CS.T.01 - Identify hardware and software problems that may occur during everyday use, then develop, apply, and explain strategies for solving these problems.	Troubleshooting - Everyday	N/A
5.NI.NCO.01 - Model and Explain how information is broken down into smaller pieces, transmitted as packages through multiple devices over networks and the internet and reassembled at the destination.		Code.org video BrainPop Video
5.NI.C.01 - Discuss real-world cybersecurity problems and identify and implement appropriate strategies for how personal information can be protected.	WCF Cyber Security Awareness Program - #ThinkTalkTeach - Internet Safety for Children (English)	Lesson 1

Quarter 2		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.DA.S.01 - Justify the format and location for storing data based on sharing requirements and the type of information	Google Drive explanation Collaborating on Google	N/A
5.DA.CVT.01 - Organize and present collected data to highlight relationships and support a claim.	Math - Graphing Language Arts - Text Features	N/A
5.DA.IM.01 - Use data to highlight or propose relationships, predict outcomes, or communicate an idea.	Math - Graphing Language Arts - Text Features	N/A

Quarter 3		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.AP.A.01 -Using grade appropriate content and complexity, compare and refine multiple algorithms for the same task and determine which is the most appropriate.	Which method do you use to solve a problem? Why? Are there other ways? (ie. repeated addition vs multiplication)	N/A
5.AP.V.01 - Using grade appropriate content and complexity, create programs that use variables to store and modify data.		Lesson 8 Lesson 10 Lesson 11

Quarter 4		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.IC.C.01 -Give examples and explain how computing technologies have changed the world and express how those technologies influence and are influenced by cultural practices.	Social Studies - General change of technology and inventions	Code.org lesson
5.IC.SI.02 -Develop, test, and refine digital artifacts or devices to improve accessibility and usability for diverse end users.	Lessons on diversity and what it means for people who are differently-abled (ie blind, deaf, paralyzed etc) Any program that makes it easier for you to use if you have a disability. SnapNRead, Co-Writer - think about what it lets you do, that you couldn't before.	
5.IC.SLE.01 - Recognize and appropriately use public domain and creative commons media and discuss the social impact of violating intellectual property rights.		Lesson 17

4th Grade Computer Science

Purpose Statement:	Students will explain how we use computers to store and organize data and justify where they store information and for what purpose. Students will explain how we use computers ethically, how to protect information, and demonstrate appropriate behavior when interacting with classmates in a digital environment.
--------------------	--

Bolded items identify learning targets/standards that must be taught to mastery and are considered a priority. Please note, however, that all learning targets/standards in the curriculum map must still be taught, but those in bold should be given extra emphasis.

Quarter 1		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.CS.D.01 - Independently, describe how internal and external parts of computing devices function to form a system.		Video on internal and external parts of a computer
5.CS.HS.01 - Model how information is translated, transmitted, and processed in order to follow through hardware and software to accomplish tasks.	Process of input - output Music teacher teaches students how to create and develop a song. The process of how software and hardware accomplish tasks.	
5.CS.T.01 - Identify hardware and software problems that may occur during everyday use, then develop, apply, and explain strategies for solving these problems.		Lesson 17
5.NI.C.01 - Discuss real-world cybersecurity problems and identify and implement appropriate strategies for how personal information can be protected.		Lesson 4 Lesson 5 Lesson 6

Quarter 2		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.DA.CVT.01 - Organize and present collected data to highlight relationships and support a claim.	Science, Math, Language Arts	

5.DA.IM.01 -Use data to highlight or propose relationships, predict outcomes, or communicate an idea.	Science, Math, Language Arts	
5.DA.S.01 - Justify the format and location for storing data based on sharing requirements and the type of information	Sharing Pdf's, format matters, images vs text vs emoj's, screenshots, docs vs word	

Quarter 3		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.AP.A.01 -Using grade appropriate content and complexity, compare and refine multiple algorithms for the same task and determine which is the most appropriate		Lesson 11 Lesson 12 Lesson 13 Lesson 16
5.AP.C.01 - Using grade appropriate content and complexity, create programs that include sequences, events, loops, and conditionals, both individually and collaboratively.		Lesson 8 Lesson 9 Lesson 10
5.AP.M.01 - Using grade appropriate content and complexity, decompose (break down) problems into smaller, manageable sub-problems to facilitate the program development process.		Lesson 11 Lesson 13 Lesson 15 Lesson 16 Lesson 18
5.AP.M.02 - Using grade appropriate content and complexity, modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.		Lesson 11 Lesson 13 Lesson 15 Lesson 16 Lesson 18

Quarter 4		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.IC.C.02 - Develop, test, and refine digital artifacts or devices to improve accessibility and usability for diverse end users.		Lesson 17
5.IC.SI.02 - Practice grade-level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior.	Padlet:discussion board, online community, digital citizenship	
5.IC.SLE.01 - Recognize and appropriately use public domain and creative commons media		Lesson 6 Lesson 17

and discuss the social impact of violating intellectual property rights.		Lesson 18
---	--	---------------------------

5th Grade Computer Science

Purpose Statement:	Students will apply computational thinking skills to create and build basic programs to accomplish a task and debug issues to ensure that the program runs as intended.
--------------------	---

Bolded items identify learning targets/standards that must be taught to mastery and are considered a priority. Please note, however, that all learning targets/standards in the curriculum map must still be taught, but those in bold should be given extra emphasis.

Quarter 1		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.CS.HS.01 - Model how information is translated, transmitted, and processed in order to follow through hardware and software to accomplish tasks.	Process of input and output (math)	N/A
5.CS.T.01 - Identify hardware and software problems that may occur during everyday use, then develop, apply, and explain strategies for solving these problems.		Lesson 17
5.NI.C.01 - Discuss real-world cybersecurity problems and identify and implement appropriate strategies for how personal information can be protected.		Lesson 5

Quarter 2		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.DA.CVT.01 -Organize and present collected data to highlight relationships and support a claim.		Lesson 16
5.DA.IM.01 - Use data to highlight or propose relationships, predict outcomes, or communicate an idea.		Lesson 16 Lesson 17 Lesson 18
5.DA.S.01 - Justify the format and location for storing data based on sharing requirements and the type of information	Google Drive explanation Collaborating on Google	N/a

Quarter 3		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.AP.A.01 - Using grade appropriate content and complexity, compare and refine multiple algorithms for the same task and determine which is the most appropriate.		Lesson 2 Lesson 3 Lesson 7 Lesson 10 Lesson 11 Lesson 12 Lesson 13 Lesson 14 Lesson 15 Lesson 20
5.AP.M.01 through 5.AP.M.02 <ul style="list-style-type: none"> Using grade appropriate content and complexity, decompose (break down) problems into smaller, manageable sub-problems to facilitate the program development process. Using grade appropriate content and complexity, modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.		Lesson 7 Lesson 8 Lesson 12 Lesson 13 Lesson 14 Lesson 15 Lesson 20
5.AP.C.01 -Using grade appropriate content and complexity, create programs that include sequences, events, loops, and conditionals, both individually and collaboratively.		Code.org Fundamentals Course F Lesson 3 Lesson 4 Lesson 9 Lesson 18
5.AP.PD 01 through 05 <ul style="list-style-type: none"> Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences. Using grade appropriate content and complexity, observe intellectual property rights and give appropriate credit when creating or remixing programs. Using grade appropriate content and complexity, describe choices made during program development using code 		Lesson 4 Lesson 7 Lesson 20

<p>comments, presentations, and demonstrations</p> <p>Using grade appropriate content and complexity, with teacher guidance, perform varying roles when collaborating with peers during the design, implementation, and review stages of program development.</p>		
---	--	--

Quarter 4		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
5.IC.C.02 - Develop, test, and refine digital artifacts or devices to improve accessibility and usability for diverse end users.		Lesson 1 Lesson 17 Lesson 18
5.IC.SLE.01 - Seek diverse perspectives for the purpose of improving computational artifacts.	<p>Lessons on diversity and what it means for people who are differently-abled (ie blind, deaf, paralyzed etc)</p> <p>Any program that makes it easier for you to use if you have a disability. SnapNRead, Co-Writer - think about what it lets you do, that you couldn't before.</p> <p>Collaborate with other schools to survey for other perspectives and needs</p>	N/A

6th Grade Computer Science

Purpose Statement:	Students will apply computational thinking to identify and resolve problems with devices. Students will investigate and select computational tools to create data for analysis. Students will critique the use of computing in society and develop strategies for safe and ethical use.
--------------------	---

Bolded items identify learning targets/standards that must be taught to mastery and are considered a priority. Please note, however, that all learning targets/standards in the curriculum map must still be taught, but those in bold should be given extra emphasis.

Quarter 1		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
6.CS.T.01 Systematically identify, resolve, and document increasingly complex software and hardware problems with computing devices and their components.	<p>Teach students how to submit their own Help Desk tickets.</p> <p>Present a number of troubleshooting problems and have students try to fix them or solve them on their own.</p>	https://studio.code.org/s/csd1-2021

Quarter 2		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
6.DA.S.01 Represent data using multiple encoding schemes.	Represent data using jpeg, gif, png, pdf, graph, etc.	https://studio.code.org/s/csd5-2021
6.DA.CVT.01 Explore a variety of computational tools and the content of their data	<p>Use data to support a claim</p> <p>Create data to analyze information</p>	https://studio.code.org/s/csd5-2021
6.DA.IM.01 Use models and simulations to formulate, refine, and test hypotheses.	Organize data in order to create a graph.	https://studio.code.org/s/csd5-2021

Quarter 3		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
6.IC.C.01. Explain how computing impacts people's everyday activities	Fake/Real News, Argumentative, Research Writing Cyber Footprint ELA Springboard, Unit 3	https://www.commonsense.org/education/digital-citizenship/curriculum?grades=6%2C7%2C8
6.IC.SI.02 Practice grade-level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior	Cyber footprint Digital citizenship Collaboration with Google Docs, Slides, etc.	https://www.commonsense.org/education/digital-citizenship/curriculum?grades=6%2C7%2C8

Quarter 4		
WY Computer Science Standard	Suggested Integration	Suggested Lesson
6.IC.SLE.01 Using grade appropriate content and complexity, describe tradeoffs between allowing information to be public and keeping information private and secure	Social media account safety (Netflix "Social Dilemma") Cyber footprint	https://www.commonsense.org/education/digital-citizenship/curriculum?grades=6%2C7%2C8 https://www.thesocialdilemma.com/

Introduction to Computer Science

(grades 7-8)

Purpose Statement:	Students will work independently and collaboratively to collect and transform grade appropriate content, find and manipulate data, make the data more useful and secure. Students will also create and improve (test and debug) a digital program.
--------------------	--

Bolded items identify learning targets/standards that must be taught to mastery and are considered a priority. Please note, however, that all learning targets/standards in the curriculum map must still be taught, but those in bold should be given extra emphasis.

WY Computer Science Standard	Suggested Lesson
8.IC.SLE.01 - Describe tradeoffs between allowing information to be public and keeping information private and secure. <i>[Practice 7.2 Communicating About Computing]</i>	Nearpod: What's The Big Deal About Internet Privacy. Nearpod: Does It Matter Who Has Your Data. Media Smarts Lessons-TBD
8.NI.C.01 Critique physical and digital procedures that could be implemented to protect electronic data/information. <i>[Practice 7.3 Communicating About Computing]</i>	Unit 5 Code.org ~Lesson 7
8.AP.V.01 Create clearly named variables that represent different data types and perform operations on their values. <i>[Practice 5.1 & 5.2 Creating Computational Artifacts]</i>	Unit 5 Code.org ~Lesson 1 ~Lesson 8 ~Lesson 14
8.AP.C.01 Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals. <i>[Practice 5.1 and 5.2 Creating Computational Artifacts]</i>	Unit 4 ~Lessons 1, 2, 3, 4
8.AP.M.01 Decompose problems and sub-problems into parts to facilitate the design, implementation, and review of programs. <i>[Practice 3.2 Recognizing and Defining Computational Problems]</i>	Unit 4 ~Lessons 5 & 6

WY Computer Science Standard	Suggested Lesson
8.CS.HS.01 Design and refine a project that combines hardware and software components to collect and exchange data. <i>[Practice 5.1 Creating Computational Artifacts]</i>	Unit 6 -Unit 2 -Web Development -Unit 3- animation
8.AP.PD.04 Document programs in order to make them easier to follow, test, and debug.	Unit 4 ~Lesson 18, 19, 20

<i>[Practice 7.2 Communicating About Computing]</i>	
8.DA.CVT.01 Using computational tools, transform collected data to make it more useful and reliable. <i>[Practice 6.3 Testing and Refining Computational Artifacts]</i>	Unit 5 ~Lesson 13

Computer Science 1

Purpose Statement:	Students will demonstrate the basics of hardware and software and explain their interactions. Students will build and strengthen logical thinking and problem-solving skills as students develop the foundations of computer programming (in either block or text coding) and create programs to solve everyday problems.
--------------------	---

Benchmarks:

CS1.1	Students will differentiate between hardware and software in a computer system, how they interact, communicate, where and how data is stored. Students will develop troubleshooting strategies based on understanding of the hardware and software components.	Standard Reference
CS1.1.1	Explain how abstractions hide the underlying implementation details of computing systems embedded in everyday objects with real-world examples.	L1.CS.D.01
CS1.1.2	Identify, define and explain the interactions between application software, system software, and hardware layers. <i>For example, text editing software interacts with the operating system to receive input from the keyboard, convert the input to bits for storage, and interpret the bits as readable text to display on the monitor.</i>	L1.CS.HS.01
CS1.1.3	Create systematic troubleshooting strategies that others can use to identify and resolve errors.	L1.CS.T.01
CS1.1.4	Translate between different bit representations of real-world phenomena, such as characters, numbers, and images using binary.	L1.DA.S.01
CS1.1.5	Identify how data elements are organized and where data is stored and evaluate the trade-offs.	L1.DA.S.02
CS1.2	Students will create programs that solve real-world problems relevant to students using procedural programming (block or text) with control structures, Boolean logic, loops, if-else statements, user interfaces and functions.	Standard Reference
CS1.2.1	Create a prototype that uses algorithms (<i>e.g., searching, sorting, finding shortest distance</i>) to provide	L1.AP.A.01

	a possible solution for a real-world problem relevant to the student.	
CS1.2.2	Describe how artificial intelligence algorithms drive many software and physical systems.	L1.AP.A.01
CS1.2.3	Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.	L1.AP.V.01
CS1.2.4	Compare and contrast simple data structures and their uses (<i>e.g., lists, stacks, queues</i>).	L2.AP.V.01
CS1.2.5	Select specific control structures and justify the selection when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.	L1.AP.C.01
CS1.2.6	Trace the execution of loops and conditional statements, illustrating output and changes in values of named variables.	L1.AP.C.02
CS1.2.7	Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.	L1.AP.M.01
CS1.2.8	Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.	L1.AP.M.02
CS1.2.9	Plan and develop programs by analyzing a problem and/or process, developing and documenting a solution, testing outcomes, and adapting the program for a variety of users. <i>Can include pseudocode, flowcharts or other diagram that demonstrate a clear plan for solving the problem.</i>	L1.AP.PD.01
CS1.2.10	Evaluate licenses that limit or restrict use of computational artifacts when using resources such as libraries. (<i>e.g., students might consider two software libraries that address a similar need, justifying their choice based on the library that has the least restrictive license</i>).	L1.AP.PD.02
CS1.2.11	Use debugging tools and strategies to identify and fix errors in a program and document their process.	L1.AP.PD.03

CS1.3		Students will engage in collaborative behaviors with classmates, establishing roles and responsibilities for appropriate behavior to develop computational artifacts both in class and in an online community.	Standard Reference
	CS1.3.1	Design and develop computational artifacts, working in team roles, using methods and tools for collaboration.	L1.AP.PD.04 L1.IC.SI.01
	CS1.3.2	Document and record design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs. **Can include comments within the program, documentation tabs, pseudocode, flowcharts or diagrams	L1.AP.PD.05
	CS1.3.3	Practice grade-level appropriate behavior and responsibilities while participating in an online community. Students will identify and report inappropriate behavior.	L2.IC.SI.01
	CS1.3.4	Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.	L1.IC.SLE.03

CS1.4		Students will analyze how computers and programming impact our world (personal - societal), and how it affects people and situations in a variety of disciplines.	Standard Reference
	CS1.4.1	Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.	L1.IC.C.01
	CS1.4.2	Test and refine computational artifacts to reduce bias and equity deficits.	L1.IC.C.02
	CS1.4.3	Demonstrate how a given algorithm applies to problems across disciplines.	L1.IC.C.03

Webpage Design 1

Purpose Statement:	Students will analyze the structure of existing websites and use HTML and CSS to create custom websites that will display information and include interactive user interfaces. Students will test multiple prototypes that address accessibility for a variety of users.
--------------------	--

Benchmarks:

CVE.WEB1.1	Students will demonstrate basic skills of internet and network functions, identifying the parts of a network and explain how those parts work together. Students will describe relationships in a network and how those relationships affect the scalability, reliability and security.	Standard Reference
CVE.WEB1.1.1	Explain the parts of a network, how they work together, protocols and their function and how information is broken down into packets & delivered	8.NI.NCO.01 (6-8 Grade Band)
CVE.WEB1.1.2	Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing.	L1.NI.NCO.01
CVE.WEB1.1.3	Give examples to illustrate how sensitive data can be affected by malware and other attacks.	L1.NI.C.01

CVE.WEB1.2	Students will create custom websites using HTML and CSS that will display information and interactive user interfaces. Students will test multiple prototypes and include designs that address accessibility for a variety of users.	Standard Reference
CVE.WEB1.2.1	Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.	L1.AP.C.03
CVE.WEB1.2.2	Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.	L1.AP.M.01
CVE.WEB1.2.3	Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.	L1.AP.M.02
CVE.WEB1.2.4	Use debugging tools to identify and fix errors in a program.	L1.AP.PD.03

	CVE.WEB1.2.5	Design and develop computational artifacts, working in team roles, using collaborative tools.	L1.AP.PD.04
	CVE.WEB1.2.6	Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs.	L1.AP.PD.05
	CVE.WEB1.2.7	Evaluate and refine computational artifacts to make them more usable and accessible.	L1.AP.PD.06
	CVE.WEB1.2.8	Modify an existing program to add additional functionality and discuss intended and unintended implications (e.g., breaking other functionality).	L2.AP.PD.07

CVE.WEB1.3		Students will analyze how computers and programming impact our world (personal - societal), and how it affects people and situations in a variety of disciplines.	Standard Reference
	CVE.WEB1.3.1	Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.	L1.IC.C.01
	CVE.WEB1.3.2	Test and refine computational artifacts to reduce bias and equity deficits.	L1.IC.C.02
	CVE.WEB1.3.3	Evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society.	L2.IC.C.02
	CVE.WEB1.3.4	Use tools and methods for collaboration.	L1.IC.SI.01
	CVE.WEB1.3.5	Practice grade-level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior.	L1.IC.SI.01
	CVE.WEB1.3.6	Explain the privacy concerns related to the collection and generation of data through automated processes that may not be evident to users.	L1.IC.SLE.02
	CVE.WEB1.3.7	Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.	L1.IC.SLE.03
	CVE.WEB1.3.8	Using grade level appropriate content and complexity, discuss the legal, social, and ethical impacts associated with software development and use, including both positive and malicious intent.	L1.IC.SLE.04

Mobile App Development

Purpose Statement:	Students will design and program mobile apps for multiple platforms that address current problems/needs. Students will include interactive user interfaces with accessibility features for diverse users into their designs.
--------------------	--

Benchmarks:

CS.MAD.1	Students will explain how abstractions can be applied to mobile app objects and will develop troubleshooting strategies based on understanding of the hardware and software components.	Standard Reference
CS.MAD.1.1	Explain how abstractions hide the underlying implementation details of computing systems embedded in everyday objects.	L1.CS.D.01
CS.MAD.1.2	Develop guidelines that convey systematic troubleshooting strategies that others can use to identify and resolve errors.	L1.CS.T.01

CS.MAD.2	Students will design and program mobile apps for multiple platforms. Students will include interactive user interfaces with accessibility features for diverse users.	Standard Reference
CS.MAD.2.1	Create a prototype that uses algorithms (e.g., searching, sorting, finding shortest distance) to provide a possible solution for a real-world problem relevant to the student.	L1.AP.A.01
CS.MAD.2.2	Describe how artificial intelligence algorithms drive many software and physical systems.	L1.AP.A.02
CS.MAD.2.3	Develop an artificial intelligence algorithm to play a game against a human opponent or solve a real-world problem.	L2.AP.A.02
CS.MAD.2.4	Use lists to simplify solutions, generalizing computational problems instead of repeatedly using simple variables.	L1.AP.V.01
CS.MAD.2.5	Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.	L1.AP.C.01
CS.MAD.2.6	Design and iteratively develop computational artifacts for practical intent, personal expression, or	L1.AP.C.03

	to address a societal issue by using events to initiate instructions.	
CS.MAD.2.7	Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.	L1.AP.M.01
CS.MAD.2.8	Construct solutions to problems using student-created components, such as procedures, modules, and/or objects.	L2.AP.M.01
CS.MAD.2.9	Create artifacts by using procedures within a program, combinations of data and procedures, or independent but interrelated programs.	L1.AP.M.02
CS.MAD.2.10	Plan and develop programs by analyzing a problem and/or process, developing and documenting a solution, testing outcomes, and adapting the program for a variety of users.	L1.AP.PD.01
CS.MAD.2.11	Plan and develop programs that will provide solutions to a variety of users using a software life cycle process.	L2.AP.PD.01
CS.MAD.2.12	Use version control systems, integrated development environments (IDEs), and collaborative tools and practices (<i>e.g., code documentation in a group software project</i>).	L2.AP.PD.02
CS.MAD.2.13	Use debugging tools to identify and fix errors in a program.	L1.AP.PD.03
CS.MAD.2.14	Develop programs for multiple computing platforms.	L2.AP.PD.03
CS.MAD.2.15	Design and develop computational artifacts, working in team roles, using collaborative tools.	L1.AP.PD.04
CS.MAD.2.16	Evaluate key qualities of a program through a process such as a code review (<i>e.g., qualities could include correctness, usability, readability, efficiency, portability, and scalability</i>).	L2.AP.PD.04
CS.MAD.2.17	Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs.	L1.AP.PD.05
CS.MAD.2.18	Develop and use a series of test cases to verify that a program performs according to its design specifications.	L2.AP.PD.05
CS.MAD.2.19	Evaluate and refine computational artifacts to make them more usable and accessible.	L1.AP.PD.06

	CS.MAD.2.20	Modify an existing program to add additional functionality and discuss intended and unintended implications (<i>e.g., breaking other functionality</i>).	L2.AP.PD.07
--	-------------	--	-------------

	CS.MAD.3	Students will analyze how computers and programming impact our world (personal - societal), and how it affects people and situations in a variety of disciplines and adapt programs based on intellectual property laws and ethical considerations.	Standard Reference
	CS.MAD.3.1	Evaluate the beneficial and harmful effects that computational artifacts and innovations have on society.	L2.IC.C.01
	CS.MAD.3.2	Test and refine computational artifacts to reduce bias and equity deficits.	L1.IC.C.02
	CS.MAD.3.3	Demonstrate how a given algorithm applies to problems across disciplines.	L1.IC.C.03
	CS.MAD.3.4	Predict how computational innovations that have revolutionized aspects of our culture might evolve	L2.IC.C.03
	CS.MAD.3.5	Use tools and methods for collaboration.	L1.IC.SI.01
	CS.MAD.3.6	Practice grade-level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior.	L2.IC.SI.02
	CS.MAD.3.7	Explain the beneficial and harmful effects that intellectual property laws can have on innovation.	L1.IC.SLE.01
	CS.MAD.3.8	Debate laws and regulations that impact the development and use of software and technology.	L2.IC.SLE.01
	CS.MAD.3.9	Explain the privacy concerns related to the collection and generation of data through automated processes that may not be evident to users.	L1.IC.SLE.02
	CS.MAD.3.10	Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.	L1.IC.SLE.03
	CS.MAD.3.11	Using grade level appropriate content and complexity, discuss the legal, social, and ethical impacts associated with software development and use, including both positive and malicious intent.	L1.IC.SLE.04

Cybersecurity

Purpose Statement:	Students will demonstrate principles of digital citizenship and cyber hygiene. Students will develop skills in cryptography, software security, networks and IT infrastructure. Students will recommend various security measures, compare and explain trade-offs and ethical impacts.
--------------------	--

Benchmarks:

CS.CS.1		Students will apply concepts of network security and evaluate cybersecurity measures, considering trade-offs in implementation, efficiency, and ethical impacts.	Standard Reference
	CS.CS.1.1	Evaluate the scalability and reliability of networks, by describing the relationship between routers, switches, servers, topology, and addressing.	L1.NI.NCO.01
	CS.CS.1.2	Describe the issues that impact network functionality (<i>e.g., bandwidth, load, latency, topology</i>).	L2.NI.NCO.01
	CS.CS.1.3	Give examples to illustrate how sensitive data can be affected by malware and other attacks.	L1.NI.C.01
	CS.CS.1.4	Compare ways software developers protect devices and information from unauthorized access.	L2.NI.C.01
	CS.CS.1.5	Recommend cybersecurity measures to address various scenarios based on factors such as efficiency, feasibility, and ethical impacts.	L1.NI.C.02
	CS.CS.1.6	Compare various security measures, considering trade-offs between the usability and security of a computing system.	L1.NI.C.03
	CS.CS.1.7	Explain trade-offs when selecting and implementing cybersecurity recommendations.	L1.NI.C.04
	CS.CS.1.8	Design and develop computational artifacts, working in team roles, using collaborative tools.	L1.AP.PD.04
	CS.CS.1.9	Explain security issues that might lead to compromised computer programs.	L2.AP.PD.06

CS.CS.2		Students will evaluate laws and regulations around software development and the internet, their impact on privacy, innovation and ethics.	Standard Reference
	CS.CS.2.1	Evaluate the ways computing impacts personal, ethical, social, economic, and cultural practices.	L1.IC.C.01

	CS.CS.2.2	Evaluate the beneficial and harmful effects that computational artifacts and innovations have on society.	L2.IC.C.01
	CS.CS.2.3	Evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society.	L2.IC.C.02
	CS.CS.2.4	Predict how computational innovations that have revolutionized aspects of our culture might evolve.	L2.IC.C.03
	CS.CS.2.5	Use tools and methods for collaboration.	L1.IC.SI.01
	CS.CS.2.6	Practice grade-level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior.	L2.IC.SI.01
	CS.CS.2.7	Explain the beneficial and harmful effects that intellectual property laws can have on innovation.	L1.IC.SLE.01
	CS.CS.2.8	Debate laws and regulations that impact the development and use of software and technology.	L2.IC.SLE.01
	CS.CS.2.9	Explain the privacy concerns related to the collection and generation of data through automated processes that may not be evident to users.	L1.IC.SLE.02
	CS.CS.2.10	Using grade level appropriate content and complexity, discuss the legal, social, and ethical impacts associated with software development and use, including both positive and malicious intent.	L2.IC.SLE.02
	CS.CS.2.11	Evaluate the social and economic implications of privacy in the context of safety, law, or ethics.	L1.IC.SLE.03
	CS.CS.2.12	Using grade level appropriate content and complexity, discuss the legal, social, and ethical impacts associated with software development and use, including both positive and malicious intent.	L1.IC.SLE.04

Computer Science 2

Purpose Statement:	Students will categorize the roles of operating system software and explain how computers facilitate logic, input, output and storage. Students will build and strengthen logical thinking and problem-solving skills as they develop skills of computer programming and create programs to solve everyday problems. Students will analyze metadata from programs and create visual representations to find patterns and explain how data is used to make decisions and learn about our world.
--------------------	--

Benchmarks:

CS2.1	Students will explain the roles of hardware and software components, operating systems, how they interact and malfunction.	Standard Reference
CS2.1.1	Categorize the roles of operating system software.	L2.CS.HS.01
CS2.1.2	Identify how hardware components facilitate logic, input, output, and storage in computing systems, and their common malfunctions.	L2.CS.T.01

CS2.2	Students will create interactive and visual data representations to analyze metadata from programs to explain how data is used to make decisions and learn about our world.	Standard Reference
CS2.2.1	Create interactive data representations using software tools to help others better understand real-world phenomena (<i>e.g., paper surveys and online data sets</i>).	L1.DA.CVT.01
CS2.2.2	Use data analysis tools and techniques to identify patterns in data representing complex systems.	L2.DA.CVT.01
CS2.2.3	Select data collection tools and techniques, and use them to generate data sets that support a claim or communicate information.	L2.DA.CVT.02
CS2.2.4	Create computational models that represent the relationships among different elements of data collected from a phenomenon or process.	L1.DA.IM.01
CS2.2.5	Formulate, refine, and test scientific hypotheses using models and simulations.	L2.DA.IM.01

CS2.3		Students will create programs that solve real-world problems relevant to students using procedural and object-based programming (text) with control structures, Boolean logic, loops, if-else statements, user interfaces and functions.	Standard Reference
	CS2.3.1	Critically examine and trace classic algorithms. Use and adapt classic algorithms to solve computational problems (<i>e.g., selection sort, insertion sort, binary search, linear search</i>).	L2.AP.A.01
	CS2.3.2	Evaluate algorithms (<i>e.g., sorting, searching</i>) in terms of their efficiency, correctness, and clarity.	L2.AP.A.03
	CS2.3.3	Compare and contrast simple data structures and their uses (<i>e.g., lists, stacks, queues</i>).	L2.AP.V.01
	CS2.3.4	Justify the selection of specific control structures when tradeoffs involve implementation, readability, and program performance, and explain the benefits and drawbacks of choices made.	L1.AP.C.01
	CS2.3.5	Trace the execution of recursion, illustrating output and changes in values of named variables.	L2.AP.C.01
	CS2.3.6	Trace the execution of loops and conditional statements, illustrating output and changes in values of named variables.	L1.AP.C.02
	CS2.3.7	Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.	L1.AP.C.03
	CS2.3.8	Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.	L1.AP.M.01
	CS2.3.9	Construct solutions to problems using student-created components, such as procedures, modules, and/or objects.	L2.AP.M.01
	CS2.3.10	Analyze a large-scale computational problem and identify generalizable patterns that can be applied to a solution.	L2.AP.M.02
	CS2.3.11	Demonstrate code reuse by creating programming solutions using libraries and APIs.	L2.AP.M.03
	CS2.3.12	Plan and develop programs by analyzing a problem and/or process, developing and documenting a	L1.AP.PD.01

		solution, testing outcomes, and adapting the program for a variety of users.	
	CS2.3.13	Plan and develop programs that will provide solutions to a variety of users using a software life cycle process.	L2.AP.PD.01
	CS2.3.14	Use version control systems, integrated development environments (IDEs), and collaborative tools and practices (<i>e.g., code documentation</i>) in a group software project.	L2.AP.PD.02
	CS2.3.15	Use debugging tools to identify and fix errors in a program.	L1.AP.PD.03
	CS2.3.16	Design and develop computational artifacts, working in team roles, using collaborative tools.	L1.AP.PD.04
	CS2.3.17	Evaluate key qualities of a program through a process such as a code review (<i>e.g., qualities could include correctness, usability, readability, efficiency, portability, and scalability</i>).	L2.AP.PD.04
	CS2.3.18	Document design decisions using text, graphics, presentations, and/or demonstrations in the development of complex programs.	L1.AP.PD.05
	CS2.3.19	Develop and use a series of test cases to verify that a program performs according to its design specifications.	L2.AP.PD.05
	CS2.3.20	Modify an existing program to add additional functionality and discuss intended and unintended implications (<i>e.g., breaking other functionality</i>).	L2.AP.PD.07
	CS2.3.21	Compare multiple programming languages and discuss how their features make them suitable for solving different types of problems.	L2.AP.PD.08

CS2.4		Students will analyze how computers and programming impact our world (personal - societal), and how it affects people and situations in a variety of disciplines and adapt programs based on intellectual property laws and ethical considerations.	Standard Reference
	CS2.4.1	Evaluate the beneficial and harmful effects that computational artifacts and innovations have on society.	L2.IC.C.01
	CS2.4.2	Evaluate the impact of equity, access, and influence on the distribution of computing resources in a global society.	L2.IC.C.02

	CS2.4.3	Demonstrate how a given algorithm applies to problems across disciplines.	L1.IC.C.03
	CS2.4.4	Use tools and methods for collaboration.	L1.IC.SI.01
	CS2.4.5	Practice grade-level appropriate behavior and responsibilities while participating in an online community. Identify and report inappropriate behavior.	L2.IC.SI.02
	CS2.4.6	Using grade level appropriate content and complexity, discuss the legal, social, and ethical impacts associated with software development and use, including both positive and malicious intent.	L2.IC.SLE.02
	CS2.4.7	Using grade level appropriate content and complexity, discuss the legal, social, and ethical impacts associated with software development and use, including both positive and malicious intent.	L1.IC.SLE.04

Appendix A

Sweetwater County School District #1

Pacing Guide

Grade/Course:		Teacher:	
---------------	--	----------	--

Code	Benchmark	Time Frame	Assessment Period			
			1	2	3	4